Create a **Student Management System**. This system includes classes:

1. A class named “**Student**”. It contains some fields: String id, String name and some methods: constructors, getters, setters
2. A class named “**StudentManager**”. It uses a hashMap to store information campus name and list of student joined the campus. (Hint: campusName is key, the list of student is value)  
    class **StudentManager**{

HashMap<String, List<Student>> h;

Public StudentManager() {

// TODO: init h

}

//this method to check a campus name already contains in HashMap

Public Boolean **checkCampus**(String campusName)

{

//TODO:

}

// add campusName and list of students to h.

Public void **addStudentsOfCampus**(String campusName, List<Student> list){

//TODO: You must code based on suggestion below:

If campusname existed in HashMap h then display {

Sout(“your campus name exsited. Do you want replace list of

student in this campus? [Y/N]”)

If (user input “Y”) {   
 put input parameters to h;   
 }  
 }  
 else put input parameters to h

}

//this method to input information into h

public void **input(){**

Scanner sc=new Scanner(System.in);  
 String choice;  
 do{

Sout(“input a campus name:”);  
 String campusname=sc.nextLine();  
 List<Student> list=new ArrayList();  
   
 do{

String id, name;

Sout(“input id:”);

Sc=new Scanner(System.in);  
 id=sc.nextLine();  
 Sout(“input name:”);

Sc=new Scanner(System.in);  
 name=sc.nextLine();  
 Student stu=new Student(id,name);

list.add(stu) ;  
 Sout(“do you continue[Y/N]?:”);

Sc=new Scanner(System.in);  
 choice=sc.nextLine();

} while(choice.equals(“Y”));

addStudentsOfCampus(campusname,list);   
 sout(“do you add more campus[Y/N]?”)

Sc=new Scanner(System.in);  
 choice=sc.nextLine();

} while(choice.equals(“Y”));

}

//return the list of students belong to a given campus name

Public List<Student> getStudentByCampus(String campusName) {  
 //TODO  
 }

//this method to test getStudentByCampus

public void test1(){

String campus;

Scanner sc=new ….  
 campus =sc.nextLine();

List<Student> result= **getStudentByCampus**(campus);  
 if( result.size()>0)   
 for( Student stu: result) sout(stu);  
 else sout(“not found”);

}

//this method will add a student stu to a specified campus name

public void addStudentToCampus(String campusName, Student stu)  
 {

//TODO: code here  
 - you get List of students belong to campus name( call the method above)   
 - add stu to this list   
 }

//this method to test the method addStudentToCampus

public void test2(){  
 String campus, stuId , stuName;  
 Scanner sc=…  
 sout(“input campus”);  
 campus=sc.nextLine();

sout(“input id”);  
 stuId=sc.nextLine();

sout(“input name”);  
 stuName=sc.nextLine();

Student stu=new Student(stuId,stuName);  
 **addStudentToCampus**(campus,stu);

}

// this method to get campus name which has maximum number of student

public String **getCampusMax**(){

//TODO

}

// this method to get maximum number of student

public int **getMaxNumberStudent**(){

//TODO

}

//this method to test getCampusMax(),getMaxNumberStudent()

public void **test3**(){

call getCampusMax

call getMaxNumberStudent

}

//this method to validate a student enroll many campus (>=2)

//if existed this case return that student

//else return null

public Student checkStudentLearnCampus(String id){

//TODO

}

////this method print all Students that leaning >=2 campus

public void displayStudentsLearnManyCampus(){

//TODO: code here

}

}  
3. Create a class named “Tester”, it contains the method main to call all methods above

Menu

1. Call the method input
2. Call the method test1
3. Call the method test2
4. Call the method test 3
5. Call the method **displayStudentsLearnManyCampus**

**-----------------------------------------------------**

HashMap h

**After call case 1, h like table below**

|  |  |
| --- | --- |
| Key | Value |
| “HN” | { {“SE123”,”Lan”} ,{“Se121”,”diep”},{“SE444”,”Lieu”}} |
| “SG” | { {“SE333”,”Hung”},{“SE456”,”Minh”},{“SE678”,”Pham”},{“SE000”,”Chi”}} |
| “DN” | { {“SE444”,”Lieu”},{“SE456”,”Minh”}} |

**After case 2:**

Input campus =”HN”

The output is: { {“SE123”,”Lan”} ,{“Se121”,”diep”},{“SE444”,”Lieu”}}

**After case 3:**

Input campus: “DN”, stuid =”SE999”, stuname=”ABC”  
h likes:

|  |  |
| --- | --- |
| Key | Value |
| “HN” | { {“SE123”,”Lan”} ,{“Se121”,”diep”},{“SE444”,”Lieu”}} |
| “SG” | { {“SE333”,”Hung”},{“SE456”,”Minh”},{“SE678”,”Pham”}, {“SE000”,”Chi”}} |
| “DN” | { {“SE444”,”Lieu”},{“SE456”,”Minh”}, {SE999”,”ABC”}} |

***You can call case 2 again to test case 3***

**After case 4:**

The campus has maximum number of students: SG

The maximum number of student: 4

After case 5:

Students which learning more campus: {“SE444”,”Lieu”}